



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/897,552	07/02/2001	James A. Sievert	RA 5202 (CST 1028.1120101	9692
27516	7590	12/14/2005	EXAMINER	
UNISYS CORPORATION MS 4773 PO BOX 64942 ST. PAUL, MN 55164-0942			ROCHE, TRENTON J	
			ART UNIT	PAPER NUMBER
			2193	

DATE MAILED: 12/14/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

Application No.

09/897,552

Applicant(s)

SIEVERT, JAMES A.

Examiner

Trenton J. Roche

Art Unit

2193

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 31 August 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 1-8 and 10-25 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 1-3, 10-12 and 25 is/are allowed.
- 6) ☒ Claim(s) 4, 5, 7, 8 and 13-24 is/are rejected.
- 7) ☒ Claim(s) 6 is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 02 July 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_.
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_.

### DETAILED ACTION

1. This office action is responsive to communications filed 31 August 2005.
2. Per Applicant's request, newly added claim 25 has been entered. Claims 1-8 and 10-25 are currently pending.
3. Claims 1-8 and 10-25 have been examined.

### *Claim Rejections - 35 USC § 101*

4. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 7 and 8 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The invention as disclosed in claims 7 and 8 is directed to non-statutory subject matter. The claimed invention as a whole must accomplish a practical application. That is, it must produce a "useful, concrete and **tangible** result." (State Street Bank & Trust Co. v. Signature Financial Group Inc., 149 F.3d at 1373, 47 USPQ2d at 1601-02.)

Specifically, claims 7 and 8 are directed to an object adapted for persistent storage, the object having a smart pointer, wherein the smart pointer includes various attributes, operations and methods. However, an object being *adapted for* (emphasis added) persistent storage does not necessarily require that the object is tangibly embodied on a computer-readable medium. Note MPEP 2106, which states:

Language that suggests or makes optional but does not require steps to be performed or does not limit a claim to a particular structure does not limit the scope of a claim or claim limitation. The following are examples of language that may raise a question as to the limiting effect of the language in a claim:

- (A) statements of intended use or field of use,
- (B) "adapted to" or "adapted for" clauses,
- (C) "wherein" clauses, or
- (D) "whereby" clauses.

This list of examples is not intended to be exhaustive. >See also MPEP § 2111.04.<

Art Unit: 2193

As such, the claim is directed purely to an object capable of being adapted for storage, and as such, essentially amounts to nothing more than the claiming of a data structure. Consequently, claims 7 and 8 are non-statutory for at least the reason that it is not tangibly embodied in a manner so as to be executable.

On this basis, claims 7 and 8 are rejected under 35 U.S.C. § 101.

***Claim Rejections - 35 USC § 112***

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 7 and 8 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. The language of claim 7 raises some confusion as to which object various sections of the claim is referring to. An initial object is referred to in line 1 of the claim, while alternate objects are introduced in lines 3 and 4. Furthermore, the phrase “object being pointed to” does not clearly outline which of the introduced objects this is referring to, as the objects of lines 1, 3 and 4 could all be separate objects, each of which has an associated pointer. As such, the claim language of independent claim 7 and its dependent claim 8 is ambiguous, and the scope of the claims cannot be reasonably ascertained.

***Claim Rejections - 35 USC § 102***

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

Art Unit: 2193

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

8. Claims 4, 5, 7, 8, 13-17 and 19-23 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent 6,125,364 to Greef et al. (hereinafter "Greef")

**Per claim 4:**

Greef discloses:

- a method for writing a plurality of objects in non-persistent storage to persistent storage  
(“Figure 7 is a flowchart demonstrating ‘storing’ objects on a flexible nonvolatile or persistence memory” in col. 5 lines 49-50)
- the objects having pointers to objects, unique object identifiers, and object types as attributes  
(“Each entity in the system has a unique object identifier and a class identifier...objects have lists of smart pointers...” in col. 4 lines 21-41)
- providing one or more common interfaces that are used by each of the plurality of objects to write the objects from non-persistent storage to persistent storage (“The two primary classes for streaming objects into and out of persistence is the Data Base Cursor and the Data Store...” in col. 4 lines 42-43. The classes are common to the objects.)
- grouping together said objects into type sets, wherein each of said objects in each of said type sets have the same type, wherein each of said type sets have a set population equal to a total number of objects inhabiting said type set (“Objects can be stored individually or grouped with other objects” in col. 2 lines 16-17. Further, the sets would inherently have a

Art Unit: 2193

set population equal to the total number of objects inhabiting the set, as the set population is a direct reflection of what is contained in the set; they would be identical.)

- counting each of said type sets and arriving at a total number of sets (“The class count field...” in col. 6 line 45. As the Data Store )
- converting each of said objects to a persistable form including obtaining a persistable form for each of said pointers to objects by obtaining a unique object identifier corresponding to each of said pointers to objects (fill the DataCursor’s buffer with class member data, class identifiers and class data offsets...” in col. 5 lines 58-59)
- writing said total number of type sets to persistent storage (Note Figure 8 and the corresponding sections of the disclosure.)
- writing each of said type sets to persistent storage (Note Figure 8 and the corresponding sections of the disclosure.)

substantially as claimed.

**Per claim 5:**

Greef discloses:

- a method for storing and restoring user objects to persistent storage (“Figure 7 is a flowchart demonstrating ‘storing’ objects on a flexible nonvolatile or persistence memory” in col. 5 lines 49-50)
- providing one or more common interfaces that are used by the user objects to store and restore the objects to/from persistent storage (“The two primary classes for streaming objects into and out of persistence is the Data Base Cursor and the Data Store...” in col. 4 lines 42-43. The classes are common to the objects.)

Art Unit: 2193

- creating a persistence controller object for managing the persistence of the user objects, the persistence controller being derived from at least one of the common interfaces (Note Figure 2 and the corresponding sections of the disclosure)
- providing a plurality of user defined classes, the classes derived from a common object base class (Note Figure 2 and the corresponding sections of the disclosure)
- creating a plurality of instances of user objects belonging to the user defined classes (Each of these classes must also be able to create a new instance of its type..." in col. 5 lines 33-34)
- providing a stream-in method and a stream-out method for each of the user defined classes ("streaming objects into and out of persistence is the Data Base Cursor and the Data Store..." in col. 4 lines 42-43)
- registering each added user defined class and added user object in a registry (Note Figure 2 and the corresponding sections of the disclosure, wherein the system is using a database. Further, "the data base will only save dirty entities..." in col. 4 lines 47-48. The database serves as a registry.)
- grouping the objects according to class (Note Figure 4 and the corresponding sections of the disclosure)
- Storing the grouped user objects to persistent storage using the stream-out methods (Note Figure 7, item 101. The DataStore directs the data cursor to storage the objects.)
- Loading the stored objects from storage into memory using the stream-in methods (Note Figure 9, item 301. The DataStore directs the data cursor to fetch the objects.)
- Registering the user objects in the registry ("the data base will only save dirty entities..." in col. 4 lines 47-48. The database serves as a registry.)

substantially as claimed.

**Per claim 7:**

Greef discloses:

- an object adapted for persistent storage, the object having a smart pointer, wherein the smart pointer includes an address attribute for containing the address of an object, and an object unique identifier attribute for containing the unique identifier of an object (Note Figure 1, the Entity Smart Pointer element. Further, “When an operation is invoked on a smart pointer, the object that is points to...” in col. 4 lines 52-54. Finally, “Each entity has a corresponding smart pointer” in col. 4 lines 36-37. As a smart pointer constitutes an object in the system, and all objects in a computer system must contain an address in the computer system, then the smart pointer, while having an identifier for the object it points to, also has an associated address in the computer system for itself.)
- wherein the object smart pointer has an assignment operation which stores the address of the object being pointed to and the unique identifier of the object being pointed to, and wherein the object includes a load method for using the smart pointer unique identifier attribute to determine and load a new smart pointer address attribute after the object being pointed to is loaded from persistent storage (“A smart pointer is then created and the smart pointer asks the Class that corresponds to the object’s class identifier, to create an object of the class.” in col. 4 lines 58-61. As all entities have smart pointers in the system, the newly created object will have a smart pointer of its own.)

substantially as claimed.

**Per claim 8:**



Art Unit: 2193

The rejection of claim 7 is incorporated, and further, Greef discloses a stream-out method for streaming out the smart pointer address attribute and unique identifier attribute as claimed (“All objects manipulate smart pointers. Objects have lists of smart pointers...” in col. 4 lines 39-40. Further, “streaming objects into...persistence...” in col. 4 lines 42-43. Since all objects have smart pointers, and the system of Greef is streaming objects into persistent storage, then the smart pointer attributes must be streamed to persistent storage as well.)

**Per claim 13:**

Greef discloses:

- a method for writing a plurality of objects in non-persistent storage to persistent storage (“Figure 7 is a flowchart demonstrating ‘storing’ objects on a flexible nonvolatile or persistence memory” in col. 5 lines 49-50)
- providing one or more common interfaces that are used by each of the plurality of objects to write the objects from non-persistent storage to persistent storage (“The two primary classes for streaming objects into and out of persistence is the Data Base Cursor and the Data Store...” in col. 4 lines 42-43. The classes are common to the objects.)
- each of the common object interfaces having a corresponding common object class (Note Figure 2 and the corresponding sections of the disclosure)
- providing a Persistent Object Registry for maintaining a database of objects to be saved to persistent storage, wherein the Persistent Object Registry is in communication with the persistent controller object, and the persistent controller object is derived from one or more of the common object classes (Note Figure 2 and the corresponding sections of the

Art Unit: 2193

disclosure, wherein the system is using a database. Further, “the data base will only save dirty entities...” in col. 4 lines 47-48. The database serves as a registry.)

- providing a first save method to save all objects in the Persistent Object Registry to persistent storage (Note Figure 7, item 101 and the corresponding section of the disclosure)

substantially as claimed.

**Per claim 14:**

The rejection of claim 13 is incorporated, and further, Greef discloses the plurality of objects being derived from one or more common object classes as claimed (Note Figure 5 and the corresponding sections of the disclosure)

**Per claim 15:**

The rejection of claim 13 is incorporated, and further, Greef discloses a unique object ID as claimed (“unique object identifier...” in col. 4 lines 21-22)

**Per claim 16:**

The rejection of claim 15 is incorporated, and further, Greef discloses objects having pointers to other objects, and identifying and recording the unique object ID of the other objects as claimed (“Persistent-Object-A has a reference to an instance of Persistent-Object-B...” in col. 5 lines 13-14)

**Per claim 17:**

Art Unit: 2193

The rejection of claim 16 is incorporated, and further, Greef discloses saving the recorded unique object Ids of the other objects as claimed (Note the rejection of claim 16. For the relationship to be preserved, the reference to Object-B from Object-A must be recorded.)

**Per claim 19:**

Greef discloses:

- a method for reading a plurality of objects in non-persistent storage to persistent storage  
(Note Figure 9 and the corresponding sections of the disclosure.)
- providing one or more common interfaces that are used by each of the plurality of objects to load the objects from persistent storage to non-persistent storage (“The two primary classes for streaming objects into and out of persistence is the Data Base Cursor and the Data Store...” in col. 4 lines 42-43. The classes are common to the objects.)
- each of the common object interfaces having a corresponding common object class (Note Figure 2 and the corresponding sections of the disclosure)
- providing a Persistent Object Registry for maintaining a database of objects to be saved to persistent storage, wherein the Persistent Object Registry is in communication with the persistent controller object, and the persistent controller object is derived from one or more of the common object classes (Note Figure 2 and the corresponding sections of the disclosure, wherein the system is using a database. Further, “the data base will only save dirty entities...” in col. 4 lines 47-48. The database serves as a registry.)
- providing a load method to save all objects in the Persistent Object Registry to persistent storage (Note Figure 9 and the corresponding section of the disclosure)

substantially as claimed.

Art Unit: 2193

**Per claims 20 and 21:**

The rejection of claim 19 is incorporated, and further, note the rejection regarding claims 14 and 15, respectively.

**Per claim 22:**

The rejection of claim 21 is incorporated, and further, note the rejection regarding claim 16.

**Per claim 23:**

The rejection of claim 22 is incorporated, and further, note the rejection regarding claim 17.

Similarly, the relationship of the objects would have to be preserved, so the pointed to object ID's would get loaded upon loading Object-A or Object-B.)

***Claim Rejections - 35 USC § 103***

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. Claims 18 and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,125,364 to Greef et al, hereafter referred to as Greef, in view of prior art of record, "Dr. GUI on Components, COM, and ATL", hereafter referred to as Dr. GUI, further in view of U.S. Patent 5,682,536 to Atkinson et al., hereafter referred to as Atkinson.

**Per claim 18:**

The rejection of claim 13 is incorporated, and further, while Dr. GUI discloses the use of the IUnknown interface, neither Greef nor Dr. GUI disclose an IPersistFile or IPersistStream interface. Atkinson discloses in an analogous system for persistently storing objects that the use of the IPersistFile and IPersistStream interfaces were well known in the art at the time the invention was made. (Note col. 66 line 35 to col. 71 line 13) It would have been obvious to one of ordinary skill in the art at the time the invention was made to utilize the IPersistFile and IPersistStream interfaces so that one could store and restore objects in the system persistently.

**Per claim 24:**

The rejection of claim 19 is incorporated, and further, note the rejection regarding claim 18.

***Allowable Subject Matter***

11. Claim 6 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

12. Claims 1-3, 10-12 and 25 are allowed.

13. The following is a statement of reasons for the indication of allowable subject matter:

The closest found prior art of record, specifically, U.S. Patent 6,125,364 to Greef et al., taken alone or in combination, fails to teach or reasonably suggest a method for creating a plurality of objects

Art Unit: 2193

from data in persistent storage in accordance with independent claim 1. Specifically, Greef does not disclose or reasonably suggest, taken alone or in combination, at least *for each object in said type set, creating an object from said data in persistent storage; for each object pointer in said objects, obtaining the unique object identifier corresponding to said object pointer, and for each obtained unique object identifier, obtaining the object address corresponding to said unique object identifier and setting each of said object pointers to said corresponding object addresses.* (Claim 1). Note pages 13-17 of Applicant's Remarks.

Regarding claims 10-11, Greef does not disclose or reasonably suggest, taken alone or in combination, at least *providing a first save method to save all objects in the Persistent Object Registry to persistent storage, providing a second save method for saving the attributes of teach class having objects to be saved to persistent storage, wherein the second save method is called by the first save method; providing a first load method for loading all objects saved in a file in persistent storage; providing a second load method for loading the attributes of each class having objects to be loaded from persistent storage, wherein the second load method is called by the first load method; registering the objects to be saved with the Persistent Object Registry using the persistent object controller, including storing the class ID and object ID of the objects to be saved; writing the objects to be saved to persistent storage using the first save method and second save method; reading the objects stored from persistent storage using the first load method and second load method...* (Claim 10). Note pages 27-30 of Applicant's Remarks.

Regarding claim 12, Greef does not disclose or reasonably suggest, taken alone or in combination, at least *creating a first object having a first object type, a first object address, and a first unique object identifier, and storing said first unique object identifier, address, and type in said object registry; creating a second object having a second object type, a second object address, and a second unique object identifier, and storing said second unique object identifier, address, and type in said object registry, said second object having a pointer attribute set equal to said first*

Art Unit: 2193

*object address; providing said second object pointer attribute to said object registry and obtaining said first object unique identifier corresponding to said second object pointer attribute in return; writing said second object to persistent storage as second object data, and writing said first object unique identifier corresponding to said second object pointer attribute to persistent storage, such that said written first object unique identifier is associated with said second object pointer attribute in persistent storage...reading said first object registry with said first object unique identifier and obtaining said first object address in return, and setting said second object pointer attribute equal to said first object address, such that said second object pointer attribute again points to said first object. (Claim 12). Note pages 30-34 of Applicant's Remarks.*

Regarding claim 25, Greef does not disclose or reasonably suggest, taken alone or in combination, at least *providing one or more Component Object Model (COM) interfaces that are used by the user objects to store and restore the objects to/from persistent storage, the one or more COM interfaces including a Persistent Object Interface derived from the IPersistStream class, and a Persistent Controller Interface derived from the IPersistFile class of the Component Object Model (COM); creating a COM Persistence Controller object for managing the persistence of the user objects, the COM Persistence Controller object being derived from the Persistent Controller Interface; providing a plurality of user defined classes, the classes derived either directly or indirectly from the Persistent Object Interface...providing a stream-in method and a stream-out method for each of the user defined classes. (Claim 25)*

### ***Response to Arguments***

14. Applicant's arguments filed 16 November 2004 concerning claims 1-3, 6, and 10-12 have been fully considered and are persuasive. The rejection of claims 1-3, 6 and 10-12 has been

Art Unit: 2193

withdrawn. Arguments concerning claims 4, 5, 7, 8 and 13-24 have been considered but they are not persuasive.

**Per claim 4:**

15. The Applicant states that Greef does not disclose providing one or more common interfaces that are used by each of the plurality of objects to write the objects from non-persistent storage to persistent storage. The Applicant further argues that Greef does not teach utilization of IPersistFile, IUnknown, and IPersistStream classes. In response to applicant's argument that the references fail to show the use of these specific COM classes, it is noted that these features upon which applicant relies are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). As noted in the prior rejection, the Data Base Cursor and the Data Store are commonly utilized classes for streaming objects into and out of persistence, and as such, qualify as common interfaces used by each object according to the broadest reasonable interpretation of the claim language.

The Applicant further argues that Greef does not disclose grouping objects together in type sets of the same type, and further, that Greef does not disclose the claimed counting step. As noted in col. 5 lines 36-44, Greef discloses objects and classes being grouped according to hierarchy. Each object hierarchy contains a number of classes. As classes are objects, and these classes are being grouped under Persistent-Object-A, Persistent-Object-B and Persistent-Object-C, then the objects (classes) are being grouped together with various other objects (classes). Furthermore, each of the Persistent-



Art Unit: 2193

Object-A, Persistent-Object-B and Persistent-Object-C objects are counting each of the types and arriving at a total number, represented by the hierarchy counts.

The rejection of claim 4 is proper and maintained.

**Per claim 5:**

The Applicant states that Greef does not disclose providing one or more common interfaces similarly to claim 4 above. Note the response to the arguments with regard to claim 4. Further, the Applicant states that the DataBase Cursor and the DataStore objects do not appear to manage the persistence of objects. In response, it is noted that col. 4 lines 44-46 discloses “The data cursor is an abstraction for objects to get and put their members to and from persistence.” The Examiner interprets this to show that the data cursor clearly has a part in the “management” of persistence of objects, and as such, anticipates the claim language. The rejection of claim 5 is proper and maintained.

**Per claims 7 and 8:**

The Applicant states that the smart pointers of Greef do not disclose including an object unique identifier as well as an address attribute for containing the address of an object. Note the modified rejection of claim 7 *supra*. The rejection of claim 7 and 8 is proper and maintained.

**Per claims 13-18:**

The Applicant presents similar arguments to those regarding claims 4 and 5 regarding common object interfaces, and as such, the Examiner directs the Applicant to the response concerning claims

Art Unit: 2193

4 and 5. The Applicant further states that while Greef discloses storing an object, this is clearly not disclosing the ability to store all objects in a system. The Examiner notes that Figure 4 shows multiple objects in the system of Greef. Furthermore, as was previously noted, these objects are disclosed as having a class hierarchy associated with them. Furthermore, note Figure 8, items 201 and 202, wherein a store method will keep getting called as long as the system is not at the top of the hierarchy. As such, Greef discloses the ability to store multiple objects in the system. The rejection of independent claim 13 and dependent claims 14-18 is proper and maintained.

**Per claims 19-24:**

Note the Examiner's response regarding claim 13.

***Conclusion***

16. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Trenton J. Roche whose telephone number is (571) 272-3733. The examiner can normally be reached on Monday - Friday, 9:00 am - 5:30 pm.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571) 272-3719. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2193

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Trenton J Roche  
Examiner  
Art Unit 2193

TJR

  
**KAKALI CHAKI**  
**SUPERVISORY PATENT EXAMINER**  
**TECHNOLOGY CENTER 2100**